Campus de l'Université de Montréal
C.P. 6128, succursale Centre-ville
Montréal QC H3C 3J7 Canada

Telephone : (514) 343-7575
Telecopieur : (514) 343-7121

crt@crt.umontreal.ca
www.crt.umontreal.ca

CRT-2002-03

Centre de recherche sur les transports (C.R.T.)
Centre for Research on Transportation

ON THE DESIGN OF FAULT-TOLERANT LOGICAL TOPOLOGIES IN WAVELENGTH-ROUTED NETWORKS SUPPORTING IP FLOWS

by

Antonio Nucci
Brunilde Sansò
Teodor Gabriel Crainic
Emilio Leonardi
Marco Ajmone Marsan

February 2002

103-03061/

PUBLICATION
CRT-2002-03

ON THE DESIGN OF FAULT-TOLERANT LOGICAL TOPOLOGIES IN WAVELENGTH-ROUTED NETWORKS SUPPORTING IP FLOWS*

by

Antonio Nucci[1]
Brunilde Sansò[2]
Teodor Gabriel Crainic[3]
Emilio Leonardi[1]
Marco Ajmone Marsan[1]

* This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), and by the Fonds pour la Formation de Chercheurs et l'Aide à la Recherche (FCAR).

[1] Dipartimento di Electronica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy (nucci,leonardi,ajmone@mail.tlc.polito.it)

[2] Département de génie électrique et génie informatique, École Polytechnique de Montréal, C.P.6079, succursale Centre-Ville, Montréal, Canada H3C 3A7 (bruni@crt.umontreal.ca)

[3] Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale Centre-ville, Montréal, Canada H3C 3J7, and Département de management et technologie, Université du Québec à Montréal, C.P. 6192, succursale Centre-ville, Montréal, Canada H3C 4R2 (theo@crt.umontreal.ca)

Dépôt légal – Bibliothèque nationale du Québec
Bibliothèque nationale du Canada, 2002

Abstract

In this paper we illustrate a new methodology for the design of fault-tolerant logical topologies in wavelength-routed optical networks exploiting wavelength division multiplexing, and supporting IP datagram flows. Our design approach generalizes the "design protection" concepts, and relies on the dynamic capabilities of IP to re-route datagrams when faults occur, thus achieving protection and restoration, and leading to high-performance cost-effective fault-tolerant logical topologies. In this paper for the first time we consider the resilience properties of the topology during the logical topology optimization process, thus extending the optimization of the network resilience also to the space of logical topologies. Numerical results clearly show that our approach outperforms previous ones, being able to obtain very effective survivable logical topologies with limited complexity.

Key words: WDM, IP, logical topology design, survivability, fault-tolerance, resilience, protection, restoration

Résumé

Nous présentons une méthodologie de design topologique qui permet de déterminer une structure logique fiable pour les réseaux de communication optiques qui exploitent le "Wavelength Division Multiplexing" et le "Waverouting" et qui supportent les modes "IP unicast" et "multicast". L'approche proposée généralise le concept de "protection structurelle" et tient explicitement compte des propriétés de routage dynamique IP lors de pannes. Cet article introduit, pour la première fois, la notion de récupération rapide ("resilience") au niveau de l'optimisation de la topologie logique. Les résultats expérimentaux indiquent que la méthode proposée est supérieure aux méthodes proposées antérieurement et qu'elle produit très efficacement des structures de réseaux de haute qualité.

Mots clés : WDM, IP, synthèse de topologie logique, réseaux optiques, tolérance aux pannes, fiabilité, récupération

1 Introduction

For the short-term implementation of a high-capacity IP infrastructure, network designers are considering the use of optical networks exploiting Wavelength Division Multiplexing (WDM) and Wavelength Routing (WR). Indeed, such networks permit the exploitation of the huge fiber capacity, with no need for complex processing functionalities in the optical domain. In WR IP networks, nodes comprise an optical section, and an electronic section; the former is an optical cross-connect (OXC), while the latter is a high-capacity IP router. Nodes are connected by optical fibers over which a WDM scheme is implemented. At each node, incoming WDM channels can either be transparently connected to outgoing channels through the OXC, possibly after wavelength convertion (but with no processing of in-transit information), or converted to the electronic domain, so that packets can be passed to the IP router, processed, and possibly retransmitted after IP routing. This setup allows the definion within the optical domain of semi-permanent optical pipes called "lightpaths" that may extend over several physical links. Thus, lightpaths can be seen as chains of physical channels through which packets are moved from a router to another toward their destinations. OXCs transparently connect the incoming WDM cannels corresponding to in-transit lightpaths, and convert to the electronic domain the incoming WDM channels corresponding to terminating lightpaths. The set of lightpaths and routers defines a logical topology, overlayed to the physical topology made of optical fibers and OXCs.

In order to best exploit the capacity of a WDM infrastructure, a crucial task is the identification of the best feasible logical topology for the transport of a given traffic pattern.

In recent years, the Logical Topology Design (LTD) problem in WDM networks was extensively studied, considering a number of different setups. It was shown that the identification of the optimal logical topology is computationally intractable for large size networks [1, 2], and several heuristic approaches were proposed for the identification of sub-optimal solutions in several different conditions [3, 4, 5].

The search for good solutions of the LTD problem can consider different objectives; in this paper we concentrate on an important characteristics of the logical topology that is produced by the LTD algorithm: its resistance to physical faults. Note that, since several lightpaths may traverse the same fiber on different wavelengths, the fault of a single physical link may cause the disruption of a number of lightpaths.

Many physical level protection and/or restoration schemes were proposed for WR networks (see [6, 7] for a survey). Such schemes always rely on the presence of spare resources in the network, that are used to restore disrupted lightpaths. In general, two disjoint physical paths p_1 and p_2 are associated with each existing lightpath l. In normal conditions, only path p_1 is used to support lightpath l; however, enough capacity is reserved on p_2 to be able to re-route lightpath l when one of the physical links belonging to p_1 fails. Protection and/or restoration schemes normally guarantee the resilience to the failure of a single physical link; thus, the capacity to be reserved in the network must be sufficient to successfully

re-route all the lightpaths disrupted by the failure of one link.

The main advantage of physical protection and/or restoration schemes, is that they entail no complex management procedure, and thus guarantee fast restorations to link failures. Their main drawback is that a large amount of network resources, in terms of both processing and bandwidth is devoted to restoration. As a consequence, physical protection and/or restoration schemes, in many situations, are not cost-effective.

In [8, 9, 10], a different fault-tolerance methodology called *design protection* was presented. In this case, restoration is obtained by exploiting the dynamic capabilities of IP routing. When a physical link fails in the optical network, the IP routing algorithm is able to update its tables, and restore disrupted paths, if the set of non-disrupted lightpaths still forms a connected topology. In order to achieve a good degree of fault resilience, it is fundamental *to map* (i.e., to route) the lightpaths onto the physical topology in such a way that, given any single physical link failure, the set of non-disrupted lightpaths still forms a connected network. Thus, an optimization of the physical mapping of lightpaths is necessary [9, 10]. In this paper we generalize the *design protection* approach, providing a powerful framework for the design of logical topologies with a good degree of fault resilience. Our approach differs from the approach proposed in [8, 9, 10], since it considers the resilience properties of the topology directly during the LTD optimization process, thus extending the optimization of the network resilience properties also to the space of the logical topologies. In addition, we further extend the *design protection* approach, also considering multicast traffic.

First, we formalize our approach aimed at the research of logical topologies with good degrees of fault resilience, providing an integer linear programming formulation which, however, is computationally intractable even for networks of moderate size; then we propose a methodology based on the application of the Tabu Search metaheuristic that produces suboptimal solutions, while limiting the computational effort.

Note that, according to the *design protection* approach, restoration requires the execution of signalling and management procedures, and thus entails significant delays; however, *design protection* is much more cost-effective with respect to physical protection schemes. Thus, *design protection* should not be necessarily considered as an alternative to physical protection, but as a complementary technique that may be successfully employed in order to reduce the level of physical protection required to achieve a desired level of fault-tolerance.

2 Problem Statement

The Fault-tolerant Logical Topology Design Problem (FLTDP) under given unicast and multicast traffic patterns can be stated as follows:

GIVEN:

i) an existing physical topology (which must be at least 2-connected), comprising nodes equipped with a limited integer number of tunable transmitters and receivers, connected by optical fibers that support a limited number of wavelengths;

ii) a description of the average traffic exchanged by sources and sets of destinations;

iii) a multi-hop routing strategy, defined both for unicast and multicast flows;

iiii) a failure scenario;

FIND

a logical topology and a "mapping" that optimize (i.e. maximize or minimize) an appropriate objective function.

2.1 Problem Formulation

In this section we report an ILP formulation of the FLTD problem. In order to limit the complexity of the notation, we restrict our formulation to networks supporting unicast traffic only. However, an ILP formulation for networks supporting multicast as well as unicast traffic can be developed in a very similar way.

2.1.1 Notation

We adopt the notational typology for multi-layered networks presented in [12]. The supra-index indicates the layer, starting by the lowest layer, zero, that represents the physical network. Let $G^0 = (V, E^0)$ be the unidirectional graph representing the physical topology. It is composed by OXC nodes V interconnected by optical fibers E^0. Let $|V| = N$ be the cardinality of set V and $|E^0| = M$ that of set E^0. Let R_i and T_i be the numbers of receivers and transmitters at physical node $i \in V$. Let S_k be the network state, where S_0 represents the no-failure state, while S_v for $v \geq 1$ is the state in which the optical fiber $v \in E^0$ is broken. Let $\mathcal{S}$ be the set of all operational states, whose cardinality is $|S| = M + 1$. Let $\mathcal{E}$ be the set of all possible lightpaths in any logical topology. Let $G^1(S_0) = (V, E^1(S_0))$ be the directed graph representing the logical topology in the no failure state. It is composed of IP routers V interconnected by lightpaths $E^1(S_0) \subseteq \mathcal{E}$. Let $G^1(S_v) = (V, E^1(S_v))$ denote the logical topology in the network state S_v, obtained from $G^1(S_0)$ by dropping all the lightpaths $u \in E^1(S_0)$ crossing over the optical fiber $v \in E^0$. Let $\Lambda = (\lambda^{sd})$ indicate the average traffic matrix where each entry λ^{sd} represents the average traffic flow between source s and destination d.

2.1.2 Decision Variables

Three types of binary variables are introduced into the formulation: X_u, $t_u^{sd}(S_k)$, Y_{uv}, that correspond, respectively, to logical topology, routing, and mapping.

The logical topology variables $X_u \in \{0,1\}$ describe the lightpaths included in the logical topology $G^1(S_0)$:

$$X_u = \begin{cases} 1, \text{ if lightpath } u \in \mathcal{E} \\ \quad \text{belongs to the logical topology } G^1(S_0) \\ 0, \text{ otherwise} \end{cases}$$

Then we can state that logical topology $G^1(S_0) = (V, E^1(S_0))$ comprises the ligthpaths $E^1(S_0) = \{u : X_u = 1, \quad u \in \mathcal{E}\}$.

The variables $t_u^{sd}(S_v) \in \{0,1\}$ contain the routing information for the logical topology $G^1(S_v)$:

$$t_u^{sd}(S_v) = \begin{cases} 1, \text{if traffic } s \to d \text{ crosses} \\ \quad \text{ligthpath } u \in \mathcal{E} \text{ in state } S_v \\ 0, \text{ otherwise} \end{cases}$$

The mapping variables $Y_{uv} \in \{0,1\}$, finally, contain the routing information of lightpaths belonging to the logical topology $G^1(S_0)$ over the physical topology G^0:

$$Y_{uv} = \begin{cases} 1, \text{ if lightpath } u \in \mathcal{E} \text{ crosses} \\ \quad \text{the optical fiber } v \in E^0 \\ 0, \text{ otherwise} \end{cases}$$

2.1.3 Mathematical Model

Let $\Gamma^+(i)$ be the set of lightpaths outgoing from node $i \in V$ and $\Gamma^-(i)$ be the set of lightpaths incoming to node $i \in V$. Let $\Theta^+(i)$ be the set of physical links outgoing from node $i \in V$ and $\Theta^-(i)$ be the set of physical links incoming to node $i \in V$. Let, finally, $O(u)$ be the origin node and $D(u)$ the destination node of lightpath $u \in \mathcal{E}$. We can then write the model constraints.

2.1.4 Constraints

- Connectivity:

$$\sum_{u \in \Gamma^+(i)} X_u \leq T_i \qquad \forall i \in V \tag{1}$$

$$\sum_{u \in \Gamma^-(i)} X_u \leq R_i \qquad \forall i \in V \tag{2}$$

where inequalities (1) indicate that the number of lightpaths outgoing from each node cannot be larger than the number of transmitters in the node, for each logical topology in the no failure state ($G^1(S_0)$); inequalities (2) indicate that the number of lightpaths incoming to each node cannot be larger than the number of receivers in the node, for each logical topology in the no failure state ($G^1(S_0)$);

- Routing

$$\sum_{u \in \Gamma^+(i)} t_u^{sd}(S_v) - \sum_{u \in \Gamma^-(i)} t_u^{sd}(S_v) = \begin{cases} 1 & \text{if } s = i \\ -1 & \text{if } d = i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$
$$\forall s, d, i \in V, \forall S_v \in \mathcal{S}$$

$$t_u^{sd}(S_v) \leq X_u \tag{4}$$
$$\forall s, d \in V, \forall S_v \in \mathcal{S}, \forall u \in \mathcal{E}$$

where equations (3) represent the routing continuity constraints for packet routes on the logical topology $G^1(S_v)$. They state that for each network operational state, an available (working) path on the logical topology must exist for each source-destination pair; equations (4), instead, state that traffic can be routed only on lightpaths belonging to the logical topology.

- Mapping:

$$Y_{uv} \leq X_u \qquad \forall u \in \mathcal{E}, v \in E^0 \tag{5}$$

$$\sum_{v \in \Theta^+(i)} Y_{uv} - \sum_{v \in \Theta^-(i)} Y_{uv} = \begin{cases} 1 & \text{if } O(u) = i \\ -1 & \text{if } D(u) = i \\ 0 & \text{otherwise} \end{cases} \tag{6}$$
$$\forall u \in \mathcal{E}, \forall i \in V$$

$$\sum_{s,d \in V} t_u^{sd}(S_v) \leq \left(\sum_{s,d \in V} \lambda_{sd} \right) (1 - Y_{uv}) \tag{7}$$
$$\forall u \in \mathcal{E}, \forall v \in E^0, \forall s, d \in V$$

where inequalities (5) ensure that only the lightpaths in the considered logical topology are mapped; iequations (6) represent routing continuity constraints for lighpaths on the physical topology G^0; inequalities (7) impose that all the lightpaths that cross the physical link v are not available in state S_v.

In order to extend the formulation to the case in which a limited number of wavelengths is available, an additional constraint must be introduced. Let W_v be the number of wavelengths supported on each fiber. The set of lightpaths $v \in E^1(S_0)$ must satisfy the following constraint:

$$\sum_{u \in \mathcal{E}} Y_{uv} \leq W_v \qquad \forall v \in E^0 \tag{8}$$

which indicates that the number of lightpaths that cross over each optical fiber has to be smaller than the wavelength number.

2.1.5 Objective Function

The objective function must be carefully selected, in order to obtain the best trade-off between network performance in normal condititions and fault resilience properties. Since the network performance depends on the network failure state, the objective function must combine the network performance levels under different network failure states.

We selected as objective of the optimization process the minimization of the maximum congestion level over all failure states:

$$\min H$$

with:

$$H \geq \left[\sum_{s,d \in V} t_u^{sd}(S_v) \lambda_{sd} \right] \qquad \forall S_v \in \mathcal{S}, \forall u \in \mathcal{E}$$

2.1.6 Observations

Note, first, that in the above formulation the routing of packets on the logical topology is unspecified, thus the minimization of the maximum congestion level over all failure states is jointly performed on all admissible logical topologies and routings.

Note, also, that the above ILP model provides a logical topology that can tolerate any single link failure. Indeed, the obtained logical topology is surely connected under any single link failure (equations 3).

Lightpath capacity constraints are ignored in the above formulation, for the sake of the model simplicity; we notice, however, that the minimization of the maximum congestion level corresponds to a minimization of the lightpath capacity needed to guarantee an efficient transport of the offered traffic. Thus, a minimization of the maximum congestion level leads to the minimization of the capacity needed to guarantee good performance.

3 Solution Strategy

The FLTD Problem is NP-Hard, since it is a generalization of the traditional LTD problem that was proved to be NP-hard. Even for moderate size networks, an optimal solution of the FLTD problem appears to be quite problematic due to the large number of variables and contraints involved in the formulation. Thus, the developement of heuristic solution methodologies is required. The approach to the FLTD problem solution problem in [8] and [14] consists in decomposing the whole problem in two independent subproblems: the Logical Topology Design (LTD) problem, in which the logical topology optimization is performed on the basis of the congestion level in the full operational state (S_0), thus ignoring the resilence property of the solution; and the Fault-tolerant Mapping (FM) problem, according to which the mapping of the logical topology lightpaths on the physical topology is aimed at the achievement of good resilience properties.

While the LTD problem has been widely investigated in the literature, and many algorithms have been proposed ([2, 3, 4, 5]), the FM problem has been considered only recently. In [8] a heuristc approach based on the application of the Tabu Search optimization algorithm has been proposed, while in [14] an ILP formulation of the FM problem is provided, and the problem is optimally solved for moderate size networks by applying the branch and cut algorithm.

In this paper we adopt a different strategy for the solution of the FLTD Problem. We apply Tabu Search for the optimization of the logical topology, considering both the case of no faults, and all possible cases of a single fault in the network. For each considered logical topology, lightpaths are routed over the physical topology, and the number of wavelengths to be used on each fiber is computed. Both lightpath routing and wavelength assignemt are obtained through a new heuristic algorithm, called HDAP, and described in Section 4. The traffic routing on the lightpaths forming the logical topology is taken to be shortest path for unicast traffic, and result from the SCTF algorithm proposed in [11] for multicast traffic. For completeness, the multicast routing algorithm is reported below.

SCTF algorithm

Let o be the source of multicast traffic, O be a temporaneous set of nodes used for the building of the steiner tree, and D the set of destinations to be reached. Let $\mathcal{T}$ be a dynamic structure in which the Steiner Tree is gathered:

Step 0: set $O = \{o\}$; $\mathcal{D} = D$; $\mathcal{T}$=NULL;

Step 1: select the node $d \in \mathcal{D}$ closest to nodes in O;

Step 2: select o', the node in O closest to d;

Step 3: extract d from $\mathcal{D}$;

Step 4: select shortest path route r from o' to d;

Step 5: add the link of r to $\mathcal{T}$;

Step 6: insert all the nodes crossed by r in O;

Step 7: if $\mathcal{D} \neq \emptyset$ GOTO Step 1, else STOP.

The algorithm requires at each step the evaluation of all the shortest paths from nodes in O to nodes in D. To reduce the algorithm complexity, it is possible to associate a weight w with each node in O, and consider only the k nodes with highest weight when selecting the closest destination d to O.

3.1 FLTDP Under Shortest-Path Routing

In order to restrict the optimization to act only on the set of the admissible logical topologies with shortest path routing, we need to introduce some extra variables and constraints.

Let us introduce an extra set of variables $\tau_u^{sd}(S_v) \in \{-1, 0, 1\}$, that represent a possible alternative routing with respect to the routing specified by t_u^{sd} on the logical topology:

$$\tau_u^{sd}(S_v) = \begin{cases} 1, \text{ if traffic } s \rightarrow d \text{ is re-routed on} \\ \quad \text{ligthpath } u \in \mathcal{E} \text{ in state } S_v \\ -1, \text{ if traffic } s \rightarrow d \text{ is no longer routed on} \\ \quad \text{ligthpath } u \in \mathcal{E} \text{ in state } S_v \\ 0, \text{ otherwise} \end{cases}$$

$\tau_u^{sd}(S_v)$ must satisfy the following constraints:

$$\sum_{u \in \Gamma^+(i)} \tau_u^{sd}(S_v) - \sum_{u \in \Gamma^-(i)} \tau_u^{sd}(S_v) = 0 \quad \forall s, d, i \in V, \forall S_v \in \mathcal{S} \tag{9}$$

$$\tau_u^{sd}(S_v) \leq X_u - Y_{uv} - t_u^{sd}(S_v) \quad \forall s, d \in V, \forall u \in \mathcal{E} \forall v \in E^0 \tag{10}$$

$$\tau_u^{sd}(S_0) \leq X_u - t_u^{sd}(S_0) \quad \forall s, d \in V, \forall u \in \mathcal{E} \tag{11}$$

$$-\tau_u^{sd}(S_v) \leq t_u^{sd} \quad \forall s, d \in V, \forall u \in \mathcal{E}, \forall S_v \in \mathcal{S} \tag{12}$$

where equations (9) represent the routing continuity constraints for the re-routing on logical topology $G^1(S_v)$. Equations (10) and (11), instead, state that traffic can be re-routed only on a path consisting of working lightpaths belonging to the logical topology; equations (12), finally, state that, after re-routing, routes defined by t_u^{sd} may be no longer valid.

Finally, we define G as:

$$G = - \sum_{\substack{S_v \in S \\ u \in \mathcal{E} \\ s,d \in V}} \tau_u^{sd}(S_v)$$

Note that G represents the difference between the total path length before re-routing and after re-routing. It is possible to find a set of τ_u^{sd} such that G assumes negative values whenever the set of t_u^{sd} does not describe a shortest path routing. In conclusion, if and only if the set t_u^{sd} defines a shortest path routing we find $\min G = 0$; thus, selecting:

$$\min[\epsilon H + \max G]$$

as objective function, where $\epsilon < 1/\sum_{sd} \lambda_{sd}$, we obtain the result of restricting the optimization to the set of logical topologies implementing a shortest path routing. Indeed, we observe that, by construction, $0 < \epsilon H < 1$, while $\max G$ can assume only integer non positive values. Thus, the minimum cost solution (i.e., the solution that minimizes $\epsilon H + \max G$) is by contruction the solution that minimizes H among those for which $\max G = 0$, i.e., it results the logical topology implementing a shortest path routing that minimizes the maximum congestion level.

Note, however, that the objective function, in this case, is non-linear, thus, since the formulation falls in the class of integer non linear programming problems, no general methodologies and tools are available for an optimal solution of the problem. For this reason, the developement of on-purpose powerful heuristic (sub-optimal) approaches is needed. In this paper we propose a sub-optimal solution mehodology based on the application of Tabu Search. We will show in the results section that our approach significantly otperforms the prevously proposed approaches, since the resilence properties of the topology are considerent during the logical topology optimization.

4 Mapping Between Physical and Logical Topology: HDAP

The definition of algorithms that optimally map the lightpaths on the physical topology is an important sub-problem of FLTDP. This problem is related to inequalities (5) and (6) in Section 2. The mapping problem can be stated as follows: given a logical topology, find a routing for each lightpath of the logical topology over the physical topology, such that a single optical link failure leaves the virtual topology connected.

This problem was recently studied in [8, 14]. In [8] this problem was found NP-complete, and Tabu Search was proposed to find a sub-optimal solution. An ILP formu-

lation of the problem was, instead, provided and solved in [14] for networks of moderate size, applying the CPLEX [15] optimization tool. However, since the mapping problem is only a part of FLTDP, the utilization of a computationally expensive algorithm to solve the mapping problem could have a disruptive impact on the CPU time necessary for the solution of FLTDP. Thus, for the solution of the mapping problem, we present a simple greedy algorithm, the Heuristic Disjoint Alternate Path (HDAP), whose computational complexity is small. A brief description of the HDAP algorithm follows.

HDAP algorithm

Let $OR(i)$ and $IR(i)$ be the sets of already routed lightpaths, respectively outgoing from and incoming to node i, and let $ON(i)$ and $IN(i)$ be the sets of outgoing and incoming lightpaths not yet routed. Let V_{ij} be the lightpath belonging to the logical topology, with endpoints i and j. Let $\mathcal{O}$ denote a set of nodes. Initialize $\mathcal{O}$ to the set of all nodes in the network:

Step 0: route all lightpaths V_{ij} whose endpoints are adjacent in the logical topology. Insert V_{ij} in $(OR(i), IR(j))$ and remove V_{ij} from $(ON(i), IN(j))$;

Step 1: if $\mathcal{O} = \emptyset$ STOP, otherwise select $i \in \mathcal{O}$ and remove i from $\mathcal{O}$;

Step 2: if $ON(i) = \emptyset$ GOTO Step3, otherwise select each $V_{ik} \in ON(i)$ and try to find a route for V_{ik} which is physically disjoint from the routes on which the $V_{ij} \in OR(i)$ and the $V_{jk} \in IR(k)$ have already been routed. If a physically disjoint route for V_{ik} has not been found, V_{ik} is routed on the shortest path. If also the shortest path is not available, due to the lack of free wavelengths, lightpath V_{ij} is not mapped.

Step 3: if $IN(i) = \emptyset$ GOTO Step1, select each $V_{ki} \in IN(i)$ and try to find a route for V_{ki} which is physically disjoint from the routes on which the $V_{ji} \in IR(i)$ and the $V_{kj} \in OR(k)$ have already been routed. If a physically disjoint route for V_{ki} has not been found, V_{ki} is routed on the shortest path. If also the shortest path is not available, due to the lack of free wavelengths, lightpath V_{ji} is not mapped.

An example of mapping produced by HDAP is shown in Fig. 1. The lightpaths that are mapped first over the physical topology are those whose end points are two adjacent physical nodes (see for example the lightpaths $1 \rightarrow 2$, $2 \rightarrow 3$, $3 \rightarrow 2$ etc..). Then, starting from node 1, the steps 2 and 3 of HDAP are iteratively and sequentially applied to all nodes of the network. Focusing on node 1, HDAP maps the outgoing remaining lightpath $1 \rightarrow 6$ over the optical fibers $(1, 3)$ and $(3, 6)$. Note that all the possible routes for lightpath $1 \rightarrow 6$ must comprise fiber $(1, 3)$, since lightpath $1 \rightarrow 2$ is already routed on fiber $(1, 2)$. Concerning the incoming lightpaths of node 1, HDAP maps lightpath $4 \rightarrow 1$ over the optical fibers $(4, 2)$ and $(2, 1)$. It is interesting to look at the mapping for lightpath $6 \rightarrow 1$. All the paths outgoing from node 6 must cross optical fiber $(6, 3)$, since $(6, 5)$ has been

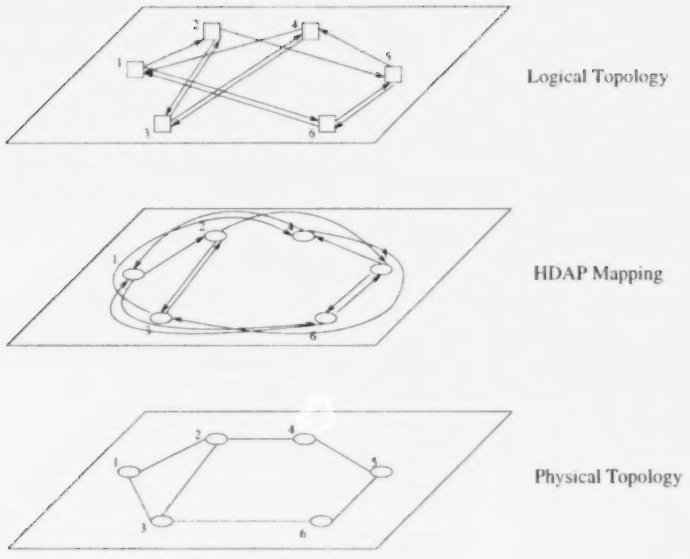

Figure 1: Example of HDAP mapping with $W_v = 3 \quad \forall v \in E^0$ and $T_i = R_i = 2 \quad \forall i \in V$.

already used by lightpath $6 \rightarrow 5$. The paths incoming to node 1 must use optical fiber $(3, 1)$ since $(2, 1)$ has been already used by lightpath $4 \rightarrow 1$. Then the only possible physical path for lightpath $6 \rightarrow 1$ is represented by the sequence of optical fibers $(6, 3)$ and $(3, 1)$. Note that if no disjoint path for $6 \rightarrow 1$ were possible, the algorithm would have selected one shortest path. It is worth to notice that if the maximum number of lightpaths that crosses a fiber is limited, some lightpath could be not mapped over the physical topology.

5 Tabu Search for FLTDP: TabuFLTDP

5.1 General Description of Tabu Search

The heuristic we propose for the solution of FLTDP relies on the application of the Tabu Search (TS) methodology. The TS algorithm can be seen as an evolution of the classical local optimum solution search algorithm called Steepest Descent (SD); however, thanks to the TS mechanism that allows worsening solutions to be also accepted, contrary to SD, TS is not subject to local minima entrapments. TS is based on a partial exploration of the space of admissible solutions, finalized to the discovery of a good solution. The exploration starts from an initial solution that is generally obtained with a greedy algorithm, and when a stop criterion is satisfied, the algorithm returns the best visited solution. For each admissible solution, a class of neighbor solutions is defined. A neighbor solution is defined as a solution that can be obtained from the current solution by applying an appropriate trans-

formation, and is also called a *move*. The set of all admissible moves uniquely defines the *neighborhood* of each solution.

At each iteration of the TS algorithm, all solutions in the neighborhood of the current one are evaluated, and the best is selected as the new current solution. Note that, in order to efficiently explore the solution space, the definition of neighborhood may change during the solution space exploration; in this way it is possible to achieve an *intensification* or a *diversification* of the search in different solution regions.

A special rule, the *Tabu list*, is introduced in order to prevent the algorithm to deterministically cycle among already visited solutions. The Tabu list stores the last accepted moves; while a move is stored in the Tabu list, it cannot be used to generate a new move. The choice of the Tabu list size is very important in the optimization procedure: too small a size could cause the cyclic repetition of the same solutions, while too large a size can severely limit the number of applicable moves, thus preventing a good exploration of the solution space.

5.2 Fundamental Aspects of TabuFLTDP

Four fundamental aspects that must be defined in TS concern:

- the choice of an initial solution
- the definition of the topology perturbation that generates the neighborhood
- the evaluation of the visited solutions
- the stop criterion

As initial solution we select the result of D-MLTDA [**?**] heuristic, which initially considers a fully-connected logical topology, and sequentially removes a set of least-loaded lightpaths from the logical topology, until the degree constraints are satisfied. To describe this algorithm, we use a bipartite graph associated with the current logical topology according to the following rules: two vertices i_n and o_n in the bipartite graph correspond to each node n in the (logical) topology; in the bipartite graph, an edge exists between i_s and o_d, whose weight is initialized to the traffic flow value between nodes s and d; a boolean variable is associated with each edge, which can assume the values Removable or Unremovable. The $R\&R$ algorithm can be described as follow:

Step 0: select the fully-connected logical topology and mark all lightpaths as Removable;

Step 1: if all the in/out-degree constraints are satisfied GOTO Step 2, else STOP;

Step 2: select o', the node in O closest to d;

Step 3: solve both the unicast and the multicast routing problems on the current topology and compute traffic flows on lightpaths;

Step 4: assign to each edge of the bipartite graph a weight equal to the flow traversing the associated lightpath;

Step 5: find a set of edges that can be removed from the graph by solving a 1-minimal Weight Matching[1] (1-mWM) on the bipartite graph. Only the edges that are marked as Removable can be chosen in the matching;

Step 6: remove all edges in the 1-minimal Weight Matching, together with the corresponding lightpath in the logical topology, only if the resulting logical topology remains connected and GOTO Step 1. If the removal of a matched lightpath would disconnect the logical topology, mark the lightpath as Unremovable.

The perturbation is defined according to the following algorithm:

- within the current solution, $2L^r$ nodes, $n_1, n_2, \ldots n_{2L^r}$, are selected, such that lightpaths $n_{2i-1} \rightarrow n_{2i},\ i = 1, 2, ..., L^r$ are present in the logical topology, and lightpaths $n_{2i+1} \rightarrow n_{2i},\ i = 1, 2, \ldots, L^r - 1$, and $n_1 \rightarrow n_{L^r}$ are absent;
- in the perturbed solution, lightpaths $n_{2i-1} \rightarrow n_{2i}$ are replaced with lightpaths $n_{2i+1} \rightarrow n_{2i},\ i = 1, 2, \ldots, L^r - 1$, and $n_1 \rightarrow n_{L^r}$.

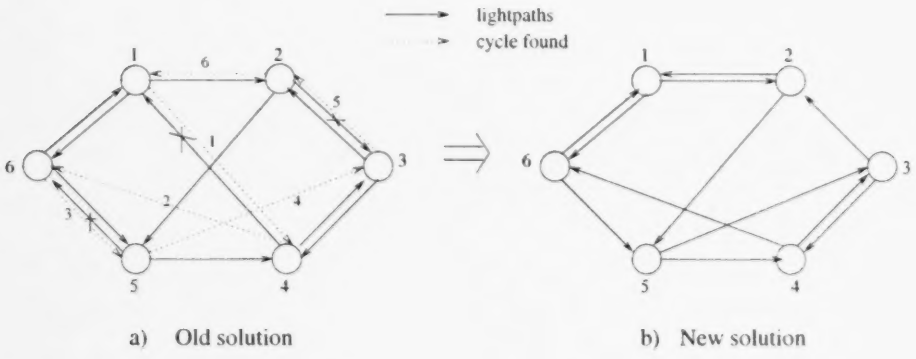

Figure 2: Perturbation using a cycle $C(1, 6)$.

This is equivalent to identifying a cycle of $2L^r$ lightpaths, L^r of which are present in the topology, while the other L^r are missing; in the identified cycle, a present lightpath if followed by a missing one, as shown in Figure 2. To obtain the neighbor topology, we have

[1]The minimal Weight Matching algorithm is similar to Maximum Weight Matching algorithm, but it provides a matching with maximal size and minimal weight.

to follow the lightpaths in the reverse direction, and remove the existing lightpaths while adding the missing ones. If $L^r = 2$, the resulting perturbation is equivalent to a well known "branch exchange" operation.

This perturbation guarantees that degree constraints are not violated, thus generating a valid move. Note that with this perturbation it is very easy and fast to implement a *diversification* and/or *intensification* criterion; we can carefully explore a region of the solution space with small cycles, and move to another region of the solution space with large cycles.

The evaluation of solutions is performed by routing the traffic on the topology for all the single link failure states S_k, and computing the maximum congestion level over all failure states.

5.3 TabuFLTDP Pseudocode

First, let us define some useful notation:

- $F_{eval}(T)$ is the evaluation function to compute the merit coefficient of logical topology T. It returns M, the maximum network congestion level over all failure states. Note that the evaluation of M requires the execution of the routing algorithm on the logical topology and of the HDAP mapping algorithm.
- *BuildInitialSolution* is used to build an initial logical topology applying $R\&R$;
- *BuildCycle*(T, l) is used to build the neighbor solution of the current solution T, using cycles of length l. When the *diversification* criterion has to be used, the cycle is longer than in normal TabuFLTDP. We denote with q the length of the normal cycle and with p the length of the cycle used for the *diversification* criterion, where $p \gg q$;
- *BuildNeighborhood*(T) is a procedure to build the neighborhood of the current solution T, by applying iteratively the *BuildCycle(T,l)* procedure;
- *BestNeighSol*(T) is a procedure that evaluates each solution in the neighborhood of T, and returns the best solution. The evaluation is based on $F_{eval}(T)$;
- $TabuList$ is a fixed size Tabu list to store the last moves;
- $\mathcal{N}(T)$ is the neighborhood of logical topology T, built applying procedure *BuildNeighborhood(T)* and using only lightpaths not belonging to $TabuList$;
- T, T^* and T^{**} represent respectively the current logical topology, the best logical topology in $\mathcal{N}(T)$, and the best solution found by FLTDP;

- $M, M^{\star}$ and $M^{\star\star}$ represent respectively the merit associated with the logical topologies $T, T^{\star}$ and $T^{\star\star}$;
- $IterationsNumber$ is the number of iterations;
- $LimitDiv$ is the number of consecutive iteration without improvments after which the $diversification\ criterion$ is applied;
- $IterBest$ is the iteration at which $T^{\star\star}$ is found.

The pseudo-code for TabuFLTDP is given in Fig. 3.

```
T:=BuildInitialSolution
IterBest := 1
T** := T
M** := M
For counter = 1 to IterationsNumber
  do
    if counter − IterBest = LimitDiv then
      T:=BuildCycle(T,p)
    N(T):=BuildNeighborhood(T)
    T*:=BestNeighSol and update TabuList
    if M* ≤ M** then
      T** := T*
      M** := M*
      IterBest := counter
    T := T*
  endfor
```

Figure 3: TabuFLTDP pseudocode

6 Complexity

We now discuss the complexity of the proposed heuristics. Let F be the number of multicast flows that must be transported in the network. Let $\Delta = T_i = R_i \quad \forall i \in V$ denote the identical in/out degrees for each node in the logical topology. For each analysed logical topology, we route i) its lightpaths over the physical topology with the HDAP algorithm, and ii) the unicast and multicast traffic over the logical topology.

The HDAP algorithm has complexity $O(N\Delta(M+\Delta Nlog(\Delta N)))$, since at most $O(N\Delta)$ iterations are executed, while at each iteration at most $O(M + \Delta Nlog(\Delta N))$ operations are required; $O(M)$ operations, indeed, are necessary to update the cost of the links of the

physical topology and $O(Nlog(\Delta N))$ operations are necessary to run the Dijkstra algorithm. Since $M \leq N^2$, the HDAP complexity is upper bounded by $O(N^3\Delta)$.

To evaluate solutions, it is necessary to route the unicast and multicast traffic. The unicast routing algorithm requires $O(N^2log(N\Delta))$ operations (Dijkstra algorithm). For the multicast traffic routing, once shortest paths are known, the computation of each Steiner Tree according to algorithm $SCTF$ requires $O(N^2k)$ operations, where k is an internal parameter of the $SCTF$ algorithm that we set to 5. In conclusion, the computation of the unicast and multicast routing requires $O(N^2(log(N\Delta) + Fk))$ operations.

The $R\&R$ heuristic to evaluate the initial solution has complexity $O(N^3(log(\Delta N) + Fk) + N^4)$, since at most $O(N)$ iterations are needed to complete the algorithm, and at each iteration it is necessary to route traffic and execute a MWM (Maximum Weight Matching) algorithm.

Let us now focus on the complexity of the TS algorithm. At each iteration, the evaluation of all solutions in the neighborhood is necessary; this requires $O(\Delta^2N^2(N^3\Delta + N^2(log(N\Delta) + Fk) + MN^2(log(N\Delta) + Fk)))$ operations, since $O(\Delta^2N^2)$ neighbors are evaluated (assuming perturbations are generated using cycles of length 4), and the evaluation of each solution requires the execution of the HDAP algorithm and the execution of the routing algorithms for each failure state (i.e., $M + 1$ times). If the number of iterations is I, the resulting complexity is $O(I\Delta^2N^2(N^3\Delta + N^2(log(N\Delta) + Fk) + MN^2(log(N\Delta) + Fk)))$. Thus, the computational complexity of TS is upper-bounded by $O(I\Delta^2N^6(log(\Delta N) + Fk))$.

7 Numerical Results

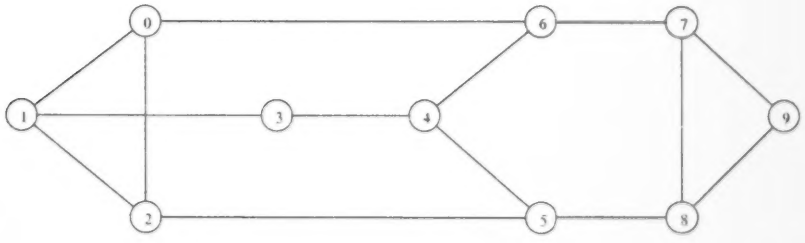

Figure 4: Physical Topology - 10 nodes 14 links

In this section we present numerical results obtained with the proposed approach (called joint optimization), and compare them against those obtained by performing a conventional optimization of the logical topology, and then optimally mapping the lightpaths on the physical topology according to the algorithm proposed in [14], and extended in order to deal with a unidirectional logical topology (this approach is called disjoint optimization).

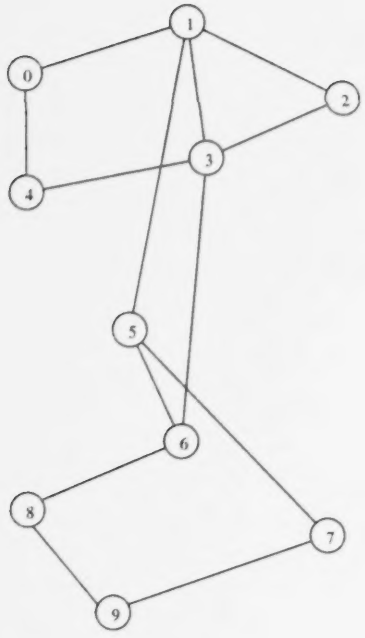

Figure 5: Physical Topology - 10 nodes 13 links

Most of the reported results refer to the two ten-nodes topologies plotted in Figs. 4 and 5, since we were unable to run the optimal mapping for larger networks. However results are also reported for larger topologies using our proposed approach. The first network that we consider was obtained by removing some nodes and links from the NSFnet topology, while the second network represents a possible Italian backbone IP network.

We consider randomly generated traffic patterns. Each source-destination traffic flow requires a bandwidth whose value is randomly extracted from an exponential distribution with mean $\mu = 1$.

The Tabu parameters used in our experiments, after an initial calibration were set as follows:

- *TabuList*: a Tabu list of fixed size equal to 7 is used;
- *Cycles size*: during the exploration of the solution space, cycles of length 4 are used. In some cases, however, different perturbation rules are used to implement the *diversification criterion*. In particular, to ease the exit from local minima regions, after 50 iterations without improvement, a cycle of size 6 is used;
- *Stop Criterion*: the procedure is stopped after a fixed number of iterations. The number of iterations is set to 300, since this value seems to provide a good trade-off

between the conflicting requirements of limiting the CPU time and obtaining good results.

In Tables 1, 2, 3 and 4 we report results obtained with the disjoint and the joint optimization techniques, for different logical network configurations on the physical topology plotted in Fig. 4. Different columns report results obtained for configuration in which different values of capacity, LC, are assigned to each lightpath.

Three important network performance indices are reported for the obtained solutions: the congestion level in the no failure state (C_{S_0}), the maximum value of the congestion level (C_S) in all the failure states, and the maximum amount of traffic (TL) that is lost in the network, due to a single link failure, expressed as percentage of the total offered traffic. Traffic losses are encountered whenever the flow on a lightpath exceeds the lightpath capacity.

We report results for different values of the nodal connectivity degree (Δ), and maximum number of wavelengths on a fiber ($\mathcal{W}$). The average performance indices (Mean) over 10 traffic instances, and the worst case (wrst) indices are reported.

In Tables 1, 2, 3 and 4 respectively, we report results for four different values of connectivity degree ($\Delta = 2, 3, 4, 5$).

It can be observed that the joint optimization approach in these cases outperforms disjoint optimization, specially for what concerns the maximum traffic lost because of failures. For example, in Table 1 we see that with disjoint optimization the maximum lost traffic is still non-null when the link capacity LC is 50, while, under joint-optimization, almost null losses are observed when the lightpath capacity is 30.

The difference between the two optimization procedures increases when the logical topology degree increases. Table 4 shows that with joint-optimization no losses are observed for configurations in which the lightpath capacity is 10, while under disjoint-optimization, losses are still registred when $LC = 25$.

Differences become even larger when the physical topology plotted in Fig. 5 is considered. In this case, trying to build a logical topology with nodal degree equal to 2, out of twenty traffic instances, ten times the disjoint optimization algorithm fails, since no mapping exists such that the logical topology remains connected under any single link failure scenario. This means that some source-destination pairs cannot communicate under some failure patterns, whichever capacity is assigned to lightpaths. In those cases, of course, the solution provided by disjoint optimization algorihms leads to unaccettable performance in terms of failure resilence. Table 5 reports results restricted to the ten cases in which the disjoint optimization does not fail.

In Tables 6, 7 and 8, results are reported for traffic scenarios comprising also multicast flows. Three multicast flows are added to the unicast traffic. For each multicast flow,

both the source and the destinations are randomly selected among all nodes in the network. The average number of destinations of each multicast flow is fixed to 6 nodes. Also multicast flows require a bandwidth whose value is randomly extracted from an exponential distribution with mean $\mu = 1$.

Tables 6, 7 and 8 refer to different values of connectivity degree ($\Delta = 2, 3, 4$ respectively). Also in presence of multicast traffic, joint-optimization significantly outperforms disjoint optimization.

Table 9 instead reports a comparison between joint optimization and disjoint optimization in terms of the required number of wavelengths. Results refer to four topologies with different number of nodes and links: the network shownd in Fig.4 (10 nodes, 14 links), the NSFnet topology (14 nodes, 21 links), the ARPAnet topology (21 nodes, 26 links) and the U.S.A. Long Distance topology (28 nodes, 45 links). We observe that while for the 10 nodes topology the logical topology resulting from disjoint-optimization was obtained by applying the optimal mapping algorithm proposed in [14], for larger networks the results were obtained by performing the heuristic HDAP mapping algorithm over the outcome of the logical topology optimization procedure because the algorithm of [14] is too complex for networks of this size. Results show that also in terms of required number of wavelengths, the application of the joint-optimization algorithm appears to be advantageous, since it yields an average saving of about 20%.

Table 10 reports the CPU time needed to run an iteration of the joint-optimization Tabu Search algorithm. All results were obtained over a 800 MHz Pentium III PC runnning Linux 6.2.

Table 10 also reports the iteration number at which the optimum solution was found; the average value over 10 instances (M O-it) and the worst case (wrst O-it) value are reported. In all cases, 300 iterations were run before stopping the algorithm. We notice that only in one istance more than 100 iterations (114) were necessary to find the optimun value.

Finally, Fig. 6 reports the average number of iterations required to find a solution that differs a given percentage from the optimum. It is worth to note that a solution that is few percentages worse then the best can be obtained in a significantly smaller number of iterations than for the best solution.

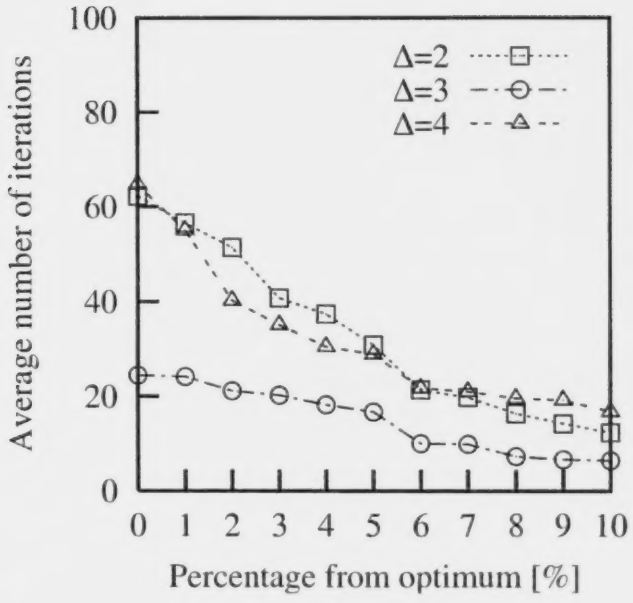

Figure 6: Average number of iterations versus the percentage distance from the optimal solution.

Parameters	**Disjoint Opt (LTDP+Opt-MP)**							
	LC=20	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*	*LC=55*
C_{S_0} *Mean*	13.02	13.02	13.02	13.02	13.02	13.02	13.02	13.02
C_S *Mean*	20.00	25.00	30.00	35.00	38.98	40.48	41.04	41.35
TL Mean [%]	27.19	20.30	13.58	7.48	2.75	1.01	0.36	0.00
C_{S_0} *wrst*	14.72	14.72	14.72	14.72	14.72	14.72	14.72	14.72
C_S *wrst*	20.00	25.00	30.00	35.00	40.00	45.00	50.00	52.81
TL wrst [%]	39.39	32.25	26.45	20.65	14.85	9.05	3.25	0.00
Parameters	**Joint Opt (FLTDP)**							
	LC=20	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*	*LC=55*
C_{S_0} *Mean*	15.96	15.96	15.96	15.96	15.96	15.96	15.96	15.96
C_S *Mean*	20.00	23.44	24.43	24.54	24.54	24.54	24.54	24.54
TL Mean [%]	6.58	1.36	0.13	0.00	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	19.84	19.84	19.84	19.84	19.84	19.84	19.84	19.84
C_S *wrst*	20.00	25.00	30.00	30.97	30.97	30.97	30.97	30.97
TL wrst [%]	18.76	7.74	1.14	0.00	0.00	0.00	0.00	0.00

Table 1: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 2$ for network 1.

Parameters	**Disjoint Opt (LTDP+Opt MP)**				
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*
C_{S_0} *Mean*	7.51	7.51	7.51	7.51	7.51
C_S *Mean*	15.00	20.00	24.46	26.85	27.7
TL Mean [%]	15.98	9.26	3.76	0.98	0.00
C_{S_0} *wrst*	9.18	9.18	9.18	9.18	9.18
C_S *wrst*	15.00	20.00	25.00	30.00	33.55
TL wrst [%]	22.16	15.72	10.56	4.75	0.00
Parameters	**Joint Opt (FLTDP)**				
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*
C_{S_0} *Mean*	9.11	9.11	9.11	9.11	9.11
C_S *Mean*	12.51	13.11	13.11	13.11	13.11
TL Mean [%]	0.15	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	11.29	11.29	11.29	11.29	11.29
C_S *wrst*	15.00	15.66	15.66	15.66	15.66
TL wrst [%]	0.76	0.00	0.00	0.00	0.00

Table 2: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 3$ for network 1.

Parameters	**Disjoint Opt (LTDP+Opt MP)**						
	LC=10	*LC=15*	*LC=20*	*LC=25*	*LC=30*	*LC=35*	*LC=40*
C_{S_0} *Mean*	6.32	6.32	6.32	6.32	6.32	6.32	6.32
C_S *Mean*	10.00	14.42	17.36	19.96	21.73	22.82	23.31
TL Mean [%]	17.42	10.36	6.90	3.85	1.83	0.57	0.00
C_{S_0} *wrst*	7.91	7.91	7.91	7.91	7.91	7.91	7.91
C_S *wrst*	10.00	15.00	20.00	25.00	30.00	35.00	39.41
TL wrst [%]	34.11	28.31	22.51	14.41	9.41	5.11	0.00
Parameters	**Joint Opt (FLTDP)**						
	LC=10	*LC=15*	*LC=20*	*LC=25*	*LC=30*	*LC=35*	*LC=40*
C_{S_0} *Mean*	7.05	7.05	7.05	7.05	7.05	7.05	7.05
C_S *Mean*	8.90	9.01	9.01	9.01	9.01	9.01	9.01
TL Mean [%]	0.22	0.00	0.00	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	8.63	8.63	8.63	8.63	8.63	8.63	8.63
C_S *wrst*	10.00	10.98	10.98	10.98	10.98	10.98	10.98
TL wrst [%]	2.04	0.00	0.00	0.00	0.00	0.00	0.00

Table 3: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 4$ for network 1.

Parameters	**Disjoint Opt (LTDP+Opt MP)**				
	LC=10	*LC=15*	*LC=20*	*LC=25*	*LC=30*
C_{S_0} *Mean*	6.07	6.07	6.07	6.07	6.07
C_S *Mean*	10	13.88	15.76	16.43	16.76
TL Mean [%]	8.98	3.61	1.14	0.38	0.00
C_{S_0} *wrst*	8.12	8.12	8.12	8.12	8.12
C_S *wrst*	10.00	15.00	20.00	25.00	27.99
TL wrst [%]	20.87	15.07	9.27	3.47	0.00
Parameters	**Joint Opt (FLTDP)**				
	LC=10	*LC=15*	*LC=20*	*LC=25*	*LC=30*
C_{S_0} *Mean*	6.27	6.27	6.27	6.27	6.27
C_S *Mean*	7.62	7.62	7.62	7.62	7.62
TL Mean [%]	0.00	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	7.48	7.48	7.48	7.48	7.48
C_S *wrst*	9.74	9.74	9.74	9.74	9.74
TL wrst [%]	0.00	0.00	0.00	0.00	0.00

Table 4: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 5$ for network 1.

Parameters	**Disjoint Opt (LTDP+Opt MP)**						
	LC=20	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*
C_{S_0} *Mean*	12.71	12.71	12.71	12.71	12.71	12.71	12.71
C_S *Mean*	20	25	30	35	38.95	41.54	42.28
TL Mean [%]	24.77	18.68	12.94	7.52	3.44	0.75	0.00
C_{S_0} *wrst*	14.23	14.23	14.23	14.23	14.23	14.23	14.23
C_S *wrst*	20.00	25.00	30.00	35.00	40.00	45.00	47.94
TL wrst [%]	35.90	27.03	19.57	13.36	8.20	3.04	0.00
Parameters	**Joint Opt (FLTDP)**						
	LC=20	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*
C_{S_0} *Mean*	14.95	14.95	14.95	14.95	14.95	14.95	14.95
C_S *Mean*	16.08	23.51	24.76	25.81	25.81	25.81	25.81
TL Mean [%]	7.27	2.92	1.07	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	18.07	18.07	18.07	18.07	18.07	18.07	18.07
C_S *wrst*	20.00	25.00	30.00	34.17	34.17	34.17	34.17
TL wrst [%]	17.30	11.68	4.31	0.00	0.00	0.00	0.00

Table 5: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 5$ for network 2.

Parameters	**Disjoint Opt (LTDP+Opt MP)**						
	LC=25	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*	*LC=55*
C_{S_0} *Mean*	13.27	13.27	13.27	13.27	13.27	13.27	13.27
C_S *Mean*	25.00	30.00	34.80	38.12	40.19	41.49	42.10
TL Mean [%]	16.81	11.71	6.90	3.74	1.75	0.50	0.00
C_{S_0} *wrst*	16.53	16.53	16.53	16.53	16.53	16.53	16.53
C_S *wrst*	25.00	30.00	35.00	40.00	45.00	50.00	53.24
TL wrst [%]	29.20	24.43	18.83	14.06	9.29	4.74	0.00
Parameters	**Joint Opt (FLTDP)**						
	LC=25	*LC=30*	*LC=35*	*LC=40*	*LC=45*	*LC=50*	*LC=55*
C_{S_0} *Mean*	16.76	16.76	16.76	16.76	16.76	16.76	16.76
C_S *Mean*	24.20	26.20	26.75	26.84	26.84	26.84	26.84
TL Mean [%]	3.66	0.88	0.09	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	21.70	21.70	21.70	21.70	21.70	21.70	21.70
C_S *wrst*	25.00	30.00	35.00	35.85	35.85	35.85	35.85
TL wrst [%]	19.41	7.89	0.81	0.00	0.00	0.00	0.00

Table 6: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 2$ for network 1 and multicast traffic.

Parameters	**Disjoint Opt (LTDP+Opt MP)**						
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*
C_{S_0} *Mean*	9.24	9.24	9.24	9.24	9.24	9.24	9.24
C_S *Mean*	15.00	20.00	24.92	29.07	31.74	32.90	33.00
TL Mean [%]	19.25	13.14	8.08	3.74	1.23	0.08	0.00
C_{S_0} *wrst*	13.03	13.03	13.03	13.03	13.03	13.03	13.03
C_S *wrst*	15.00	20.00	25.00	30.00	35.00	40.00	40.85
TL wrst [%]	28.44	23.67	17.94	10.35	5.58	0.81	0.00
Parameters	**Joint Opt (FLTDP)**						
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*	*LC=40*	*LC=45*
C_{S_0} *Mean*	10.42	10.42	10.42	10.42	10.42	10.42	10.42
C_S *Mean*	13.72	14.81	14.81	14.81	14.81	14.81	14.81
TL Mean [%]	1.62	0.00	0.00	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	12.81	12.81	12.81	12.81	12.81	12.81	12.81
C_S *wrst*	15.00	19.75	19.75	19.75	19.75	19.75	19.75
TL wrst [%]	8.28	0.00	0.00	0.00	0.00	0.00	0.00

Table 7: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 3$ for network 1 and multicast traffic.

Parameters	**Disjoint Opt (LTDP+Opt MP)**				
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*
C_{S_0} *Mean*	7.02	7.02	7.02	7.02	7.02
C_S *Mean*	15.00	18.22	20.54	21.22	21.71
TL Mean [%]	7.88	3.60	1.62	0.46	0.00
C_{S_0} *wrst*	8.97	8.97	8.97	8.97	8.97
C_S *wrst*	15.00	20.00	25.00	30.00	34.34
TL wrst [%]	23.07	13.68	8.90	4.14	0.00
Parameters	**Joint Opt (FLTDP)**				
	LC=15	*LC=20*	*LC=25*	*LC=30*	*LC=35*
C_{S_0} *Mean*	8.41	8.41	8.41	8.41	8.41
C_S *Mean*	10.35	10.35	10.35	10.35	10.35
TL Mean [%]	0.00	0.00	0.00	0.00	0.00
C_{S_0} *wrst*	10.57	10.57	10.57	10.57	10.57
C_S *wrst*	13.45	13.45	13.45	13.45	13.45
TL wrst [%]	0.00	0.00	0.00	0.00	0.00

Table 8: Comparison between FLTDP and the disjoint optimization of LTDP and Opt-MP with $\Delta = 4$ for network 1 and multicast traffic.

Mean W-Number	**Disjoint Opt (LTDP+Opt MP)**		
	$\Delta = 2$	$\Delta = 3$	$\Delta = 4$
Network 1: 10 nodes 14 links	3.2	5.0	6.1
NSFnet: 14 nodes 21 links	3.9	5.1	6.7
ARPAnet: 21 nodes 26 links	6.4	9.2	12.6
U.S.A. Long Distance: 28 nodes 45 links	7.2	11.4	14.5
Mean W-Number	**Joint Opt (FLTDP)**		
	$\Delta = 2$	$\Delta = 3$	$\Delta = 4$
Network 1: 10 nodes 14 links	2.5	3.8	5.1
NSFnet: 14 nodes 21 links	2.3	4.4	6.3
ARPAnet: 21 nodes 26 links	5.4	8.4	11.7
U.S.A. Long Distance: 28 nodes 45 links	7.0	9.2	12.1

Table 9: Number of wavelengths required to map the logical topology

CPU TIME	$\Delta = 2$			$\Delta = 3$			$\Delta = 4$	
	1-It[sec]	*M O-It*	*wrst O-It*	*1-It[sec]*	*M O-It*	*wrst O-It*	*1-It[sec]*	*M O-It*
Network 1	0.78	19.88	85	1.38	31.55	91	1.49	65.67
NSFnet	6.69	62.22	98	7.12	24.44	41	9.66	64.67
ARPAnet	60.46	36.22	67	47.48	43.22	91	46.98	40.78
U.S.A. Long Distance	246.78	36.1	60	177.20	68	114	160.43	63.5

Table 10: CPU times for one iteration of the Tabu Search algorithm.

8 Conclusions

In this paper we proposed a new methodology for the design of fault-tolerant logical topologies in wavelength-routed WDM networks supporting unicast and multicast IP datagram flows.

Our approach to protection and restoration generalizes the concepts first proposed in [8, 9, 10], and relies on the exploitation of the intrinsic dynamic capabilities of IP routing, thus leading to cost-effective fault-tolerant logical topologies.

Our approach differs from those proposed in [8, 9, 10], since it considers the resilience properties of the topology during the logical topology optimization process, thus extending the optimization of the network resilience performance also on the space of logical topologies.

Numerical results clearly show that our approach largely outperforms the previous ones, and is able to obtain very good logical topologies with fault-tolerance properties at a limited cost.

Acknowledgments

Partial funding for this project has been provided by the Natural Sciences and Engineering Council of Canada through its Research Grant program, and by the Fonds F.C.A.R. of the Province of Québec.

References

[1] D.Bienstock, O.Gueluek, "Computational Experience with a Difficult Mixed-Integer Multicommodity Flow Problem", *Mathematical Programming*, Vol.68, pp.213-235, 1995,

[2] B.Yener, T.Boult, "A Study of Upper and Lower Bounds on Minimum Congestion Routing in Lightwave Networks", *IEEE Infocom'94*, Toronto, Ontario, Canada, June 1994.

[3] E.Leonardi, M.Mellia, M.Ajmone Marsan, "Algorithms for the Topology Design in WDM All-Optical Networks", *Optical Networks Magazine, Premiere Issue*, Vol.1, n.1, pp.35–46, January 2000.

[4] R.Ramaswami, K.N.Sivarajan, "Design of Logical Topologies for Wavelength-Routed Optical Networks", *IEEE Journal of Selected Areas in Communications*, Vol.14, n.6, pp.840-851, June 1996.

[5] B.Mukherjee, D.Banerjee, S.Ramamurthy, A.Mukherjee, "Some Principles for Designing a Wide-Area WDM Optical Network", *IEEE Transactions on Networking*, Vol.4, n.5, pp. 684-695, October 1996.

[6] A. Fumagalli, L. Valcarenghi "IP Restoration vs. WDM Protection: Is there an Optimal Choice?", *IEEE Network*, pp.34-41, November/December 2000.

[7] G. Mohan, C. Siva Ram Murthy "Lightpath Restoration in WDM Optical Networks", *IEEE Network*, pp.24-32, November/December 2000.

[8] J.Armitage, O.Crochat, J.Y.Le Boudec "Design of a Survivable WDM Photonic Network", *in Proc. INFOCOM '97* pp.244-252, April 1997

[9] J O.Crochat, J.Y.Le Boudec "Design Protection for WDM Optical Networks", *IEEE Journal on Selected Areas in Communication*, Vol.16, n.7, pp.1158-1165, September 1998.

[10] J O.Crochat, J.Y.Le Boudec, O.Gerstel "Protection Interoperability for WDM Optical Networks", *IEEE Transaction on Networking*, Vol.8,n.3, pp.384-395, June 2000.

[11] S.Ramanathan, "Multicast Tree Generation in Networks with Asymmetric Links", *IEEE/ACM Transaction on Networking*, Vol.4, n.4, pp.558-568, August 1996.

[12] M.Plante, B.Sansó "A Typology for Multi-technology Multi-service Network Synthesis Methods", *GERAD Publication*, G-98-51, July 1999.

[13] M. Mellia, A. Nucci, A. Grosso, E. Leonardi, M. Ajmone Marsan "Optimal Design of Logical Topologies in Wavelength-Routed Optical Networks with Multicast Traffic", *submitted to Globecom 2001*.

[14] E. Modiano, A. Narula-Tam "Survivable routing of logical topologies in WDM networks", *in Proc. INFOCOM '01* Vol.1, pp. 348-357, Anchorage, Alaska, April 2001.

[15] `http://www.ilog.com/products/cplex/`